

REMARKS

This application has been carefully reviewed in light of the Office Action dated September 18, 2008. Claims 118 to 121, 124 to 133 and 136 to 142 are in the application. Claims 118, 130 and 142 are independent. Reconsideration and further examination are respectfully requested.

The Office Action again alleged that the executed declaration is defective for failing to provide a complete post office address for Applicants. It is believed that this objection was entered through oversight and its withdrawal is requested. As correctly noted in the Office Action, it is permissible to supply the post office address in an Application Data Sheet. Such an Application Data Sheet was included with the Preliminary Amendment filed July 23, 2008. Accordingly, since a complete post office address has been provided, reconsideration and withdrawal of the objection are respectfully requested.

If the foregoing objection to the declaration is repeated in a subsequent Office Action, the Examiner is respectfully requested to clarify how the post office address provided in the Application Data Sheet is incomplete.

Claims 118 to 121, 124 to 133 and 136 to 142 were rejected under 35 U.S.C. § 103(a) over U.S. Patent No. 5,732,277 (Kodosky) in view of <http://www.uiml.org/> (UIML), in view of U.S. Patent No. 6,968,539 (Huang) and further in view of U.S. Patent No. 5,481,741 (McKaskle). Reconsideration and withdrawal of these rejections are respectfully requested.

Referring to the claim language, Claim 118 is directed to a data processing apparatus comprising a library for storing a plurality of filters and a processor for processing a computer program stored on a computer-readable storage medium. The

processor executes, by processing the computer program, a selecting step, a modifying step, and a generating step. The selecting step selects and loads a plurality of desired filters from the library based on a first user instruction in order to form an application, wherein the plurality of selected desired filters are arranged in a sequence based on the first user instruction, and wherein the plurality of selected desired filters are used for filtering a data object based on the sequence. The modifying step modifies a user interface description object, written in a markup language, to add codes of user interface components corresponding to the selected desired filters arranged in the sequence. The generating step generates display data for displaying the user interface components corresponding to the plurality of selected desired filters in a display apparatus, by parsing the user interface description object modified in said modifying step, wherein a user inputs data to the plurality of selected desired filters via the user interface components, and wherein a user interface component corresponding to a filter further selected based on a second user instruction from the plurality of desired filters selected in said selecting step may be selectively hidden.

Amended independent Claims 130 and 142 are directed to a method and a computer-readable medium, respectively, that substantially correspond to the apparatus of Claim 118.

Applicants respectfully submit that the applied art, considered either alone or in combination, fails to disclose or suggest all of the features of the independent claims. In particular, the applied art, either alone or in combination, fails to disclose or suggest at least the feature of selectively hiding a user interface component corresponding to a filter further selected based on a second user instruction from a plurality of selected desired filters, wherein the desired filters are selected and loaded from a library based on a first

user instruction in order to form an application, are arranged in a sequence based on the first user instruction, and are used for filtering a data object based on the sequence.

As understood by Applicants, Kodosky discloses a method for programming a computer to execute a procedure based on a graphical interface which utilizes data flow diagrams to represent the procedure. The method stores a plurality of executable functions, scheduling functions, and data types. A data flow diagram is assembled in response to a user input utilizing icons which correspond to the respective executable functions, scheduling functions, and data types which are interconnected by arcs on a screen. See Kodosky, Abstract.

However, nothing in Kodosky is seen to disclose or suggest selectively hiding a user interface component corresponding to a filter further selected based on a second user instruction from a plurality of selected desired filters, wherein the desired filters are selected and loaded from a library based on a first user instruction in order to form an application, are arranged in a sequence based on the first user instruction, and are used for filtering a data object based on the sequence.

McKaskle is seen to disclose a method for providing attribute nodes in a data flow diagram which allow a user to programmatically access various parameters of a control or indicator that affect the visual output of the control or indicator on a front panel. See McKaskle, Abstract.

The Office Action alleges that McKaskle teaches selectively hiding part of a user interface that corresponds to a filter in Column 5, lines 31-34. The cited portion of McKaskle is seen to teach the programing of a block diagram to automatically hide certain controls or indicators when they are not in use. However, McKaskle is not seen to disclose or suggest hiding controls or indicators when they are in use. According to the claim

language, the selected filters are “used for filtering a data object”. Thus, unlike McKaskle, although a user interface component of a filter may be selectively hidden, the filter itself is still “used for filtering a data object”. Accordingly, like Kodosky, McKaskle is also not seen to disclose or suggest selectively hiding a user interface component corresponding to a filter further selected based on a second user instruction from a plurality of selected desired filters, wherein the desired filters are selected and loaded from a library based on a first user instruction in order to form an application, are arranged in a sequence based on the first user instruction, and are used for filtering a data object based on the sequence.

UIML and Huang have been reviewed, but are not seen to compensate for the above-noted deficiencies of Kodosky and McKaskle.

In light of these deficiencies of the applied art, Applicants submit that independent Claims 118, 130 and 142 recite subject matter that would not have been obvious from the art applied against them, and are therefore in condition for allowance.

The other pending claims in this application are each dependent from the independent claims discussed above and are therefore believed allowable for at least the same reasons. Because each dependent claim is also deemed to define an additional aspect, however, the individual consideration of each on its own merits is respectfully requested.

No other matters being raised, it is believed that the entire application is fully in condition for allowance, and such action is courteously solicited.

Applicants' undersigned attorney may be reached in our Costa Mesa,
California office at (714) 540-8700. All correspondence should continue to be directed to
our below-listed address.

Respectfully submitted,

A handwritten signature in black ink, appearing to read "Michael K. O'Neill", is written over a horizontal line.

Attorney for Applicants

Michael K. O'Neill

Registration No.: 32,622

FITZPATRICK, CELLA, HARPER & SCINTO
30 Rockefeller Plaza
New York, New York 10112-3800
Facsimile: (212) 218-2200

FCBS_WS 2777112v1